AFRL-RH-WP-TR-2016-0094

FOREIGN LANGUAGE ANALYSIS AND RECOGNITION (FLARE) FINAL REPORT

Brian M. Ore
Stephen A. Thorn
David M. Hoeferlin

SRA International
5000 Springfield Street, Suite 200
Dayton, OH, 45431

Raymond E. Slyh
Eric G. Hansen

Air Force Research Laboratory
711th Human Performance Wing
Airman Systems Directorate
Human Trust and Interaction Branch
2255 H Street
Wright-Patterson AFB, OH, 45433

OCTOBER 2016

Final Report

(STINFO COPY)

AIR FORCE RESEARCH LABORATORY
711TH HUMAN PERFORMANCE WING
AIRMAN SYSTEMS DIRECTORATE
HUMAN-CENTERED ISR DIVISION
HUMAN TRUST AND INTERACTION BRANCH
WRIGHT-PATTERSON AFB OH 45433
AIR FORCE MATERIAL COMMAND
UNITES STATES AIR FORCE

NOTICE AND SIGNATURE PAGE

Using Government drawings, specifications, or other data included in this document for any purpose other than Government procurement does not in any way obligate the U.S. Government. The fact that the Government formulated or supplied the drawings, specifications, or other data does not license the holder or any other person or corporation; or convey any rights or permission to manufacture, use, or sell any patented invention that may relate to them.

This report was cleared for public release by the 88th Air Base Wing Public Affairs Office and is available to the general public, including foreign nationals. Copies may be obtained from the Defense Technical Information Center (DTIC) (http://www.dtic.mil).

AFRL-RH-WP-TR-2016-0094 HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION IN ACCORDANCE WITH ASSIGNED DISTRIBUTION STATEMENT.

//Signed//	//Signed//
RAYMOND E. SLYH, Ph.D., WUM	LOUISE A. CARTER, Ph.D., DR-IV
Division Human Trust and Interaction Branch	Chief, Human-Centered ISR Division
Airman Systems Directorate	Airman Systems Directorate
711th Human Performance Wing	711th Human Performance Wing
Air Force Research Laboratory	Air Force Research Laboratory

This report is published in the interest of scientific and technical information exchange and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
8–10–2016	Final	1 December 2014 – 8 October 2016

4. TITLE AND SUBTITLE

Foreign Language Analysis and Recognition (FLARe) Final Report

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

[1]Brian M. Ore, Stephen A.Thorn, David M. Hoeferlin

[2]Raymond E. Slyh, Eric G. Hansen

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

H06K (5328X02S)

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

[1]SRA International
5000 Springfield Street, Suite 200
Dayton, OH 45431

[2]Air Force Research Laboratory
711 Human Performance Wing
Airman Systems Directorate
Human Trust and Interaction Branch
Wright-Patterson AFB, OH 45433

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Air Force Materiel Command
Air Force Research Laboratory
711 Human Performance Wing
Airman Systems Directorate
Human-Centered ISR Division
Wright-Patterson AFB, OH 45433

10. SPONSOR/MONITOR'S ACRONYM(S)

711 HPW/RHXS

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

AFRL-RH-WP-TR-2016-0094

12. DISTRIBUTION / AVAILABILITY STATEMENT

Distribution A: Approved for public release.

13. SUPPLEMENTARY NOTES

88ABW-2017-0506, cleared 7 February 2017.

14. ABSTRACT

This final technical report provides research results in the areas of Automatic Speech Recognition (ASR) and Information Retrieval (IR). Hybrid Deep Neural Network (DNN) Hidden Markov Model (HMM) systems were developed using i-vector input. An English ASR system was developed for the International Workshop on Spoken Language Translation (IWSLT) 2015 evaluation. ASR systems were developed for Arabic, Chinese, Farsi, Russian, and Ukrainian.

15. SUBJECT TERMS

Automatic Speech Recognition (ASR), Information Retrieval (IR).

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Raymond E. Slyh
U	U	U	SAR	31	19b. TELEPHONE NUMBER *(include area code)* N/A

Standard Form 298 (Rev. 8–98)

Prescribed by ANSI Std. Z39.18

TABLE OF CONTENTS

Section **Page**

LIST OF TABLES

SUMMARY

This document provides a summary of work completed by government researchers and SRA International under the work unit H06K (5328X02S), Foreign Language Analysis and Recognition (FLARe). This work was performed over the period 1 December 2014 to 8 October 2016 under contract FA8650-09-D-6939.

The following tasks were completed on Automatic Speech Recognition (ASR). Hybrid Deep Neural Network (DNN) Hidden Markov Model (HMM) systems were developed using i-vector input. An English ASR system was developed for the International Workshop on Spoken Language Translation (IWSLT) 2015 evaluation. ASR systems were developed for Arabic, Chinese, Farsi, Russian, and Ukrainian.

Improvements were made to the Haystack Multilingual Multimedia Information Extraction and Retrieval (MMIER) System that was initially developed under a prior work unit. Major additions to the user interface include the following: a new administrative section, updates to the file upload capability, inclusion of named entities, integration of Optical Character Recognition (OCR) results, and the addition of an HTML5-based media player. The processing pipeline was updated to provide support for hybrid DNN-HMM speech recognition systems with i-vector input, speech recognition using Kaldi, ASR system combination, OCR using Tesseract, named entity detection using the Massachusetts Institute of Technology Information Extraction (MITIE) tool, machine translation using BBN Broad Operational Language Translation (BOLT) and SDL, and text recasing using Moses.

1.0 INTRODUCTION

This document provides a summary of work completed by government researchers and SRA International under the work unit 5328X02S, Foreign Language Analysis and Recognition (FLARe). This work was performed over the period 1 December 2014 to 8 October 2016 under contract FA8650-09-D-6939.

The following tasks were completed on Automatic Speech Recognition (ASR). Hybrid Deep Neural Network (DNN) Hidden Markov Model (HMM) systems were developed using i-vector input. An English ASR system was developed for the International Workshop on Spoken Language Translation (IWSLT) 2015 evaluation. ASR systems were developed for Arabic, Chinese, Farsi, Russian, and Ukrainian.

Improvements were made to the Haystack Multilingual Multimedia Information Extraction and Retrieval (MMIER) system that was initially developed under a prior work unit. Major additions to the user interface include the following: a new administrative section, updates to the file upload capability, inclusion of named entities, integration of Optical Character Recognition (OCR) results, and the addition of an HTML5-based media player. The processing pipeline was updated to provide support for hybrid DNN-HMM speech recognition systems with i-vector input, speech recognition using Kaldi, ASR system combination, OCR using Tesseract, named entity detection using the Massachusetts Institute of Technology Information Extraction (MITIE) tool, machine translation using BBN Broad Operational Language Translation (BOLT) and SDL, and text recasing using Moses.

This report is organized as follows. Section 2.0 describes the experiments and accomplishments. Section 3.0 summarizes conclusions drawn from the experiments.

2.0 EXPERIMENTS AND ACCOMPLISHMENTS

This section discusses the experiments and accomplishments for the covered period. Section 2.1 discusses the ASR experiments that were performed, and Section 2.2 describes the improvements made to the Haystack MMIER System.

2.1 ASR Experiments

This section discusses the ASR experiments that were conducted. Section 2.1.1 describes how i-vectors were used as additional feature input for hybrid DNN-HMM speech recognition systems. Section 2.1.2 describes the English ASR system that was developed for the IWSLT 2015 evaluation campaign. Lastly, Sections 2.1.3–2.1.7 describe the Arabic, Chinese, Farsi, Russian, and Ukrainian speech recognition systems that were developed for Haystack.

2.1.1. I-Vector Input for DNNs

This section describes how i-vectors were used as additional feature input for hybrid DNN-HMM speech recognition systems. The i-vector technique was originally developed for speaker verification and is based on Joint Factor Analysis (JFA) [1]. In the i-vector paradigm, a speaker and channel dependent Gaussian Mixture Model (GMM) supervector s is defined as:

$$s = m + Tw_i \tag{1}$$

Where m is the GMM supervector from the Universal Background Model (UBM), T is the total variability matrix, and w is the i-vector. The supervector m is formed by concatenating the mean vectors from all mixture components in the UBM; the supervector s is formed in a similar manner as m from a speaker and channel dependent GMM that was adapted from the UBM; and the total variability matrix T is a rectangular matrix with low rank (typically 100–400) that can be estimated using the procedure described in [1, 2]. The remainder of this section describes the English i-vector systems developed using the ALIZE toolkit [3] and the Massachusetts Institute of Technology (MIT) LLSpeech software.

The ALIZE toolkit was used to develop an i-vector system on 166 hours of audio from 838 Technology, Entertainment, and Design (TED) talks. The data were harvested from the TED website as described in [4]. The feature set was based on the default ALIZE configuration and included 19 Mel Frequency Cepstral Coefficients (MFCCs) with 19 delta and 11 acceleration coefficients, plus delta energy. All features were extracted using HTK [5] and normalized to zero mean and unit variance on a per talk basis. The GMM-UBM included 1024 mixture components with diagonal covariance matrices, and the i-vector dimension was set to 100. Two different methods were investigated for normalizing the i-vectors: z-scoring and the Eigen Factor Radial (EFR) algorithm.

The z-score for each i-vector was calculated as:

$$w' = \frac{w - u}{\sigma} \tag{2}$$

Where μ is the mean vector and σ is the standard deviation vector estimated on the training set. The EFR algorithm estimates the mean vector μ and covariance matrix Σ from the training set and normalizes each i-vector as follows:

$$w' = \frac{\Sigma^{\frac{1}{2}}(w-u)}{\left|\Sigma^{\frac{1}{2}}(w-u)\right|} \tag{3}$$

An initial GMM-HMM speech recognition system was trained on the TED data using HTK. Phonemes were modeled using state-clustered across-word triphones, and the final HMM set included 6000 shared states with an average of 28 mixtures per state. Speaker Adaptive Training (SAT) was applied using Constrained Maximum Likelihood Linear Regression (CMLLR) transforms, and the models were discriminatively trained using the Minimum Phone Error (MPE) criterion. The feature set consisted of 13 Perceptual Linear Prediction (PLP) coefficients with mean normalization applied on a per utterance basis. Delta, acceleration, and third differential coefficients were appended to form a 52 dimensional feature vector, and Heteroscedastic Linear Discriminate Analysis (HLDA) was applied to reduce the feature dimension to 39. This system was used to generate state-level time alignments of the TED data.

Three hybrid DNN-HMM speech recognition systems were developed using Theano and a version of HTK that we modified according to the method of [6]. The baseline system was trained without i-vectors. The DNN included a context window of 9 frames on the input, a single hidden layer with 1000 logistic sigmoid units, and 6000 output units corresponding to the shared states of the GMM-HMM system. The feature set consisted of 13 PLPs with delta and acceleration coefficients, and all features were normalized to zero mean and unit variance on a per-talk basis. Cross entropy training was performed using a minibatch size of 512 and an initial learning rate of 0.008 that was adjusted according to the QuickNet newbob algorithm.[1] DNNs with i-vector input were trained by simply appending the talk specific i-vector to each set of stacked features [7]. One DNN was trained using the z-score normalized i-vectors, and a second DNN was trained using the EFR normalized i-vectors. Each system was evaluated on the tst2013 partition from the IWSLT evaluation campaign. Decoding was performed using HDecode with the trigram Language Model (LM) described in [8]. The baseline system yielded a 25.9% Word Error Rate (WER), including the z-score normalized i-vectors yielded a 24.1% WER, and including the EFR normalized i-vectors yielded a 23.7% WER. EFR normalization was used for all remaining experiments discussed in this document.

DNNs with 5 hidden layers were trained using layer growing back propagation [9]. In addition to the PLP feature input described above, networks were also trained on PLP features that were transformed using CMLLR. These systems applied a single transform per speaker. Lastly, i-vector and hybrid DNN-HMM systems were trained on 390 hours of audio from TED, Euronews [10], and HUB4 [11, 12]. These systems were trained using the same procedure described above. The final HMM set included 8000 shared states, and each DNN included 7 hidden layers with 1000 logistic sigmoid units per layer.

[1] http://www.icsi.berkeley.edu/Speech/faq/nn-train.html

Table 1 shows the WERs obtained on the tst2013 partition. Including i-vectors reduced the WER for each feature set; furthermore, the PLP with i-vector system outperformed the PLP-CMLLR feature set.

Table 1: English WER using hybrid DNN-HMM systems with i-vectors
Systems were trained on (1) TED and (2) TED, Euronews, and HUB4.

DNN Input	TED	TED Euronews HUB4
PLP	20.2	18.7
PLP-CMLLR	18.3	17.6
PLP + i-vector	17.8	16.9
PLP-CMLLR + i-vector	17.3	16.7

As an alternative to the ALIZE toolkit, an i-vector system was developed on the TED data using the MIT LLSpeech toolkit. This system used the same HTK MFCC feature set, GMM-UBM architecture, and i-vector dimension as the ALIZE system. A hybrid DNN-HMM system with non-CMLLR PLP features and i-vector input was trained and tested using the same procedure described above. A WER of 17.6% was obtained on the tst2013 partition, which is a 0.2% absolute improvement compared to the ALIZE i-vector system. All i-vector systems discussed in the remainder of this document were trained with the MIT LLSpeech toolkit.

2.1.2. IWSLT 2015

This section discusses the English ASR system that was developed for the IWSLT 2015 evaluation campaign. This task focuses on the automatic transcription of TED talks, which professionally recorded presentations are given on a variety of topics related to technology, entertainment, and design. Each talk is a maximum of 18 minutes in length. The TED website[2] makes the video recordings and closed captions from over 2200 talks available for download.

Acoustic models were trained on 1787 talks. The audio was extracted from each video file using FFmpeg,[3] and then downsampled to 16 kHz using SoX.[4] Long periods of untranscribed audio were removed from each talk using the time marks from the closed captions, and word alignments were automatically generated using an HTK GMM-HMM system developed on HUB4. These alignments were used to split each talk into utterances that were shorter than 20 seconds and included 0.1–0.25 seconds of non-speech at the end points. Next, closed caption filtering [13] was applied to the TED data to sequester utterances that may include transcription errors. Each talk was decoded using the HUB4 HMMs and a trigram LM that was estimated on the transcripts for the talk. The recognizer

[2] https://www.ted.com

[3] Available at: http://www.ffmpeg.org

[4] Available at: http://sox.sourceforge.net

outputs were compared to the transcripts, and a data partition was created using all utterances with a WER less than 30%. This process yielded 336 hours of audio.

LMs were developed on the TED data, the English Gigaword corpus [14], and the News 2007–2014 texts from the Association for Computational Linguistics Workshop on Machine Translation (WMT).[5] Cross entropy difference scoring [15] was used to select subsets of Gigaword and News 2007–2014 that matched the TED domain. Interpolated trigram and 4-gram LMs were estimated on TED, 1/4 of Gigaword, and 1/4 of News 2007–2014. A maximum entropy Recurrent Neural Network (RNN) LM was trained on the same data set using the RNNLM toolkit [16]. The network included 160 hidden units, 300 classes in the output layer, 4-gram features for the direct connections, and a hash size of 10^9. The LM vocabulary included 100000 words and was chosen using the select-vocab program from the Stanford Research Institute LM (SRILM) toolkit.

Table 2: English WER on tst2013 using hybrid DNN-HMM systems with PLP, filterbank, and LLSpeech i-vector input

DNN Input	WER
PLP	16.3
PLP + i-vector	14.0
PLP-CMLLR + i-vector	13.7
filterbank	15.1
filterbank + i-vector	13.6

A GMM-HMM speech recognition system, i-vector extractor, and hybrid DNN-HMM system were developed on the TED data using the same procedure described in Section 2.1.1. A total of five DNNs were trained using the following inputs: PLP features, PLP features with i-vectors, PLP features transformed using CMLLR with i-vectors, filterbank features, and filterbank features with i-vectors. The PLP and filterbank feature sets included 13 and 24 static coefficients, respectively. Delta and acceleration coefficients were appended to each feature set, and all features were normalized to zero mean and unit variance on a per-talk basis. The final HMM set included 8000 shared states, and each DNN included 6 hidden layers with 1024 logistic sigmoid units per layer. Decoding was performed using HDecode with the trigram LM. Table 2 shows the WERs obtained on manually segmented version of tst2013. Based on these results, the filterbank features and i-vector input were used for all DNNs described in the remainder of this section.

One method for adapting a DNN to better fit the characteristics of an individual speaker is to adjust the network weights. Adaptation data was selected for each talk using the output from the first pass recognizer. First, the recognition lattices from the hybrid DNN-HMM system were rescored with

[5] Available at: http://www.statmt.org/wmt15/translation-task.html

the interpolated 4-gram LM. Next, 1000-best lists were extracted from each lattice and rescored with the RNN LM. The final LM scores were obtained by linearly interpolating the log probabilities from the 4-gram and RNN LM. Interpolation weights of 0.25 for the 4-gram and 0.75 for the RNN were chosen based on results from previous experiments. The 1000-best lists were resorted according to the updated scores, and confidence scores were estimated at the frame level by aligning the top N hypotheses from each utterance and counting the number of matching HMM states. Next, speaker-dependent DNNs were estimated on frames with a confidence score greater than c. For each speaker, the initial DNN was updated using a learning rate of 0.001 and a single epoch of training. Table 3 shows the WERs obtained on the manually segmented version of tst2013 using different values for N and c. Based on these results, $N = 20$ and $c = 0.9$ was used for the evaluation.

Table 3: English WER on tst2013 using different N-best list sizes and thresholds c for confidence filtering

Adaptation Data	Decode	4-gram	4-gram+RNN
None	13.6	12.8	11.4
$N = 1, c = 0.0$	11.0	10.6	10.1
$N = 20, c = 0.9$	11.5	11.0	9.8
$N = 20, c = 0.8$	11.4	10.9	9.9
$N = 20, c = 0.7$	11.4	10.9	9.9
$N = 50, c = 0.9$	11.6	11.1	9.8

Whereas in the 2011–2012 IWSLT evaluations the test data were manually segmented into spoken utterances, the 2013–2015 evaluations provided each talk without timing information. Two neural network based Speech Activity Detectors (SADs) were investigated for segmenting each talk into utterances and removing long periods of non-speech. The first network was previously developed for the 2013 IWSLT evaluation [4]. The second network was trained on 80 hours of manually transcribed data from HUB4 and 5 hours of public domain music downloaded from Wikimedia Commons,[6] the United States Air Force band,[7] and the Open Goldberg Variations project.[8] The network included a context window of 41 frames on the input, 2 hidden layers of 512 neurons with logistic activation functions, and 3 output units corresponding to speech, silence/noise, and music.

The feature set consisted of 24 filterbank features with delta and acceleration coefficients, and all features were globally normalized to zero mean and unit variance. Training was performed using

[6] Available at: http://commons.wikimedia.org

[7] Available at: http://usafband.af.mil

[8] Available at: http://www.opengoldbergvariations.org

layer growing back propagation with a minibatch size of 512 and an initial learning of 0.008 that was halved after the second epoch.

Automatic segmentation of the test data was performed by evaluating the SAD, applying a Dynamic Programming (DP) algorithm to choose the best sequence of states, and then inserting an utterance boundary at non-speech segments longer than T seconds. Systems were evaluated using $T = 0.3$, $T = 0.5$, and T equal to the median pause duration for that talk. Table 4 shows the WERs obtained on the tst2013 partition using each segmentation method. Based on these results, the TED SAD with $T = 0.5$ was used for the evaluation.

Table 4: English WER on tst2013 using different methods for segmenting each talk into utterances

Segmentation	Decode	4-gram	4-gram+RNN
Manual	13.6	12.8	11.6
TED, $T = 0.3$	14.6	13.8	12.8
TED, $T = 0.5$	14.3	13.6	12.4
TED, T =median	14.4	13.5	12.4
HUB4, $T = 0.5$	15.1	14.4	13.2

The previous experiments trained DNNs with logistic sigmoid hidden units. In order to obtain the best performance, it was necessary to pretrain the networks using layer growing back propagation. Recently it has been found that Rectified Linear Units (ReLUs) yield faster convergence than logistic units, provide better generalization, and eliminate the need for pretraining [17]. The ReLU activation function is defined as $f(x) = \max(0, x)$. A DNN was trained on the TED data using the filterbank features and i-vectors as input. The network included 5 hidden layers with 1024 ReLUs per layer, and the method from [18] was used to randomly initialize the weights

$$W \sim U\left[-\beta \cdot \frac{\sqrt{6}}{\sqrt{n_i + n_{i+1}}}, \beta \cdot \frac{\sqrt{6}}{\sqrt{n_i + n_{i+1}}}\right] \quad (4)$$

Where $U[-x, x]$ is the uniform distribution over the interval $[-x, x]$, n_i is the number of units in the i^{th} layer, and $\beta = 0.5$. Note that $\beta = 4.0$ was used to initialize the weights when using logistic sigmoid hidden units. Pretraining was not applied, and the network weights were estimated using cross entropy training with a minibatch size of 512 and an initial learning rate of 0.0005 that was adjusted according to the QuickNet newbob algorithm.

A network was also trained using max-norm regularization [19], which constrains the norm of the incoming weight vector $\|w\|_2$ at each unit in layer j to be bound by a constant c_j when updating the weights. The constraints c_j were estimated by first training a DNN without max-norm regularization, and then calculating c_j as 0.8 times the average norm of all units in layer j. Table 5 shows the WERs obtained on the automatically segmented version of tst2013. The DNN with ReLUs and max-norm regularization was used for the evaluation.

Table 5: English WER on tst2013 using logistic sigmoid and ReLU hidden units with max-norm regularization

Hidden Units	Decode	4-gram	4-gram+RNN
Logistic Sigmoid	14.3	13.6	12.4
ReLU	13.9	13.3	11.9
ReLU max-norm	13.7	13.0	11.9

A second ASR system was developed using the Kaldi speech recognition toolkit [20].[9] This system was based on the LIUM recipe as released with Kaldi.[10] The training data matched what was used for the HTK system. First, a network was developed to produce bottleneck features. The network included 2 hidden layers with 1500 units per layer and a 40 dimension bottleneck layer. The input features consisted of MFCCs from 40 filterbanks and 3 pitch features. These 40 bottleneck features were then used to build a GMM-HMM system. SAT was applied using feature-space Maximum Likelihood Linear Regression (fMLLR) transforms. These models were then used to train a DNN of the Deep Belief Network (DBN) variety described as having 6 hidden layers with 2048 units per layer. Lastly, four iterations of sequence training were applied using the state-level Minimum Bayes Risk (sMBR) criterion. This system was evaluated using the trigram LM to produce recognition lattices, which were then rescored with the 4-gram and RNN LMs.

Table 6 shows the WER of each system on tst2013 after evaluating the decoder, rescoring with the 4-gram LM, and interpolating the 4-gram and RNN LM scores. The final hypotheses were selected by applying N-best Recognizer Output Voting Error Reduction (ROVER) to the output from the adapted HTK system and the Kaldi system. The combined system yielded a 9.4% WER on tst2013 and a 6.6% WER on tst2015.

[9] All Kaldi systems discussed in this document were built by Mr. Eric Hansen

[10] Available at: https://github.com/kaldi-asr/kaldi/tree/master/egs/tedlium/s5

Table 6: English WER on tst2013 using the final evaluation systems

ASR System	Decode	4-gram	4-gram+RNN
HTK first-pass	13.7	13.0	11.9
HTK adapted	11.3	10.9	10.0
Kaldi	13.3	12.6	11.4

2.1.3. Arabic

This section describes the Arabic ASR system that was developed for Haystack. Acoustic models were trained on 276 hours of speech from the Global Autonomous Language Exploitation (GALE) and Topic Detection and Tracking (TDT) [21] corpora. A GMM-HMM system, i-vector extractor, and hybrid DNN-HMM system were developed using the same procedure as the HTK first-pass system used for the IWSLT 2015 evaluation, with the following exceptions: (1) max-norm regularization was not applied, (2) the GMM-HMM PLP features were bandlimited from 125–3800 Hz, and (3) the i-vector MFCC features and DNN filterbank features were bandlimited from 0–4000 Hz. The final HMM set included 7000 shared states, and the DNN included 5 hidden layers with 1024 ReLUs per layer.

An interpolated trigram LM was estimated on GALE, TDT, and the Arabic Gigaword corpus [22] using the procedure described in [23]. Decomposition was not applied to the most frequently occurring 10000 words or any word with a stem shorter than 3 characters. The final vocabulary included 100000 tokens. This system yielded a 14.9% WER on the GALE Phase II development partition.

2.1.4. Chinese

An initial set of Chinese acoustic models was trained on 486 hours of audio from the GALE corpus. The GALE text was first segmented into words using the Linguistic Data Consortium (LDC) Chinese word segmenter. A pronunciation dictionary was created by mapping the Chinese characters to pinyin and splitting the pinyin into a 95 phoneme set that included tone markings.

Pronunciations for English words were obtained by mapping phonemes from the English Carnegie Mellon University (CMU) pronunciation dictionary[11] to the Chinese phoneme set and training a Sequitur grapheme-to-phoneme system [24].

GMM-HMM speech recognition systems were trained using three different feature sets. The first feature set included 13 PLP features, plus a log-pitch feature, with delta and acceleration coefficients. The log-pitch feature was extracted using getf0 and pitch values over unvoiced segments were defined using the method described in [25]. The second feature set included 13 PLPs with delta and acceleration coefficients, plus a log-pitch feature, a delta-log-pitch feature, and a

[11] Available at: http://www.speech.cs.cmu.edu/cgi-bin/cmudict

probability of voicing feature. The pitch features were extracted using the Kaldi toolkit, which uses a highly modified version of the getf0 method [26]. The third feature set was identical to the second set, except that PLP third differential coefficients were included and HLDA was applied to reduce the feature dimension from 55 to 42. All PLP features were bandlimited from 125–3800 Hz and mean normalization was applied on a per utterance basis. The models were trained using maximum likelihood estimation, and each HMM set included 6000 shared states with an average of 28 mixtures per state.

An interpolated trigram LM was estimated on GALE, the Chinese Gigaword corpus [27], and broadcast news transcripts from HUB4 [28]. The text was segmented into words using the LDC Chinese word segmenter, and the final vocabulary included 50168 words. The GMM-HMM models described above were evaluated on the HUB4 test partition using HDecode with the trigram LM. The following Character Error Rates (CERs) were obtained with each feature set: (1) 19.2%, (2) 17.3%, and (3) 15.3%.

Based on these results, a GMM-HMM speech recognition system was trained on 547 hours from GALE using the PLP feature set that included Kaldi pitch features and HLDA. SAT was applied using CMLLR transforms, and the models were discriminatively trained using the MPE criterion. Next, an i-vector extractor and hybrid DNN-HMM system were developed using the same procedure described in Section 2.1.3. The final HMM set included 6000 shared states, and the DNN included 5 hidden layers with 1024 ReLUs per layer. A second GMM-HMM system, i-vector extractor, and hybrid DNN-HMM system were trained on the 547 hours of GALE data, plus 147 of conversational telephone speech from the HKUST corpus. The HMM set included 10000 shared states, and the DNN included 5 hidden layers with 1024 ReLUs per layer.

These systems were evaluated on the HUB4 and HKUST test partitions. Table 7 shows the CER obtained on each test set. Whereas including the HKUST data increased the CER of the GMM-HMM system on HUB4, the additional data improved the CER of the hybrid DNN-HMM system on both test sets.

Table 7: Chinese CER on HUB4 and HKUST

	GMM-HMM		Hybrid DNN-HMM	
Training Data	HUB4	HKUST	HUB4	HKUST
GALE	10.0	57.0	6.6	43.7
GALE+HKUST	10.7	46.7	6.4	31.7

2.1.5. Farsi

This section describes the Farsi ASR system that was developed for Haystack. Acoustic models were trained on 18 hours from the Appen Mobile Network Mini Database,[12] 20 hours from the Appen Conversational Telephone corpus,[13] 17 hours from the Translation System for Tactical Use (TRANSTAC) corpus, and 19 hours from a news broadcast and broadcast conversation corpus. A GMM-HMM speech recognition system, i-vector extractor, and hybrid DNN-HMM system were developed using the same procedure described in Section 2.1.3. The final HMM set included 5000 shared states, and the DNN included 5 hidden layers with 1024 ReLUs per layer.

An interpolated trigram LM was estimated on the training transcripts from TRANSTAC and BBN, articles downloaded from Wikipedia,[14] and the Uppsala Persian corpus.[15] Pronunciations for all words were derived using the Appen Farsi Morphological Table[16] and the lexicon included with the Appen Conversational Telephone corpus. Words that were missing from the lexicon and could not be vowelized using the morphological table were pronounced using a Sequitur grapheme-to-phoneme system. This system yielded a 37.1% WER on a broadcast and broadcast conversation test set.

2.1.6. Russian

Russian acoustic models were trained on 19 hours from GlobalPhone [29], 14 hours from VoxForge,[17] and 43 hours of broadcast news that were manually transcribed in the Speech and Communication Research, Engineering, Analysis, and Modeling (SCREAM) laboratory. A GMM-HMM speech recognition system, i-vector extractor, and hybrid DNN-HMM system were developed using the same procedure described in Section 2.1.3. The final HMM set included 5000 shared states, and the DNN included 5 hidden layers with 1024 ReLUs per layer.

Interpolated trigram and 4-gram LMs were estimated on the broadcast news transcripts and the News 2007–2014 texts from WMT. A maximum entropy RNN LM was trained on the same data set using the RNNLM toolkit. The network included 160 hidden units, 300 classes in the output layer, 4-gram features for the direct connections, and a hash size of 10^9. The vocabulary included 400000 words and was chosen using the select-vocab tool from the SRILM toolkit. Pronunciations for all words were derived using the Festival speech synthesis system.

A second GMM-HMM system and hybrid DNN-HMM system were developed using word position dependent phonemes. Three positions were modeled for each phoneme: beginning of word, word internal, and end of word. Each hybrid DNN-HMM system was evaluated on the broadcast news test set using the trigram LM and a version of Sphinx-4 that we modified to read HMM state likelihoods from HTK feature files. The word position independent system yielded a 22.1% WER, and the word

[12] Product code FAR_ASR001 available at: http://catalog.appenbutlerhill.com

[13] Product code FAR_ASR002 available at: http://catalog.appenbutlerhill.com

[14] Available at: http://dumps.wikipedia.org/fawiki

[15] Available at: http://stp.lingfil.uu.se/~mojgan/UPC.html

[16] Product code FAR_MOR001 available at: http://catalog.appenbutlerhill.com

[17] Available at: http://www.repository.voxforge1.org/downloads

position dependent system yielded a 20.8% WER. Based on these results, word position dependent phonemes were used for all systems discussed in the remainder of this section.

Kaldi ASR systems were developed using the same procedure described in Section 2.1.2. One DNN system was trained on MFCC features, and a second DNN system was trained on bottleneck features. Each system was evaluated on the broadcast news test set, and LM rescoring was applied as described in Section 2.1.2. Table 8 shows the WER after evaluating the decoder, rescoring with the 4-gram LM, and interpolating the 4-gram and RNN LM scores. The final hypothesis was selected by applying N-best ROVER to the output from all three systems; this produced a 16.3% WER.

Table 8: Russian WER on the broadcast news test set

Sequence training was applied to all systems.

ASR System	Decode	4-gram	4-gram+RNN
HTK-3.5 filterbank + i-vector	19.7	19.6	18.9
Kaldi MFCC	21.1	20.9	20.3
Kaldi bottleneck	19.2	18.9	18.3

The ASR systems described so far were developed using Theano and a version of HTK-3.4.1 that we modified to support hybrid DNN-HMM systems. Recently the beta version of HTK-3.5 was released, which includes native support for hybrid DNN-HMM systems. Differences between HTK-3.5 and the Theano recipe used in the SCREAM Laboratory include the weight initialization, training vector randomization, learning rate schedule, use of momentum, use of gradient clipping, and support for lattice-based sequence training. The HNTrainSGD tool from HTK-3.5 was used to train a DNN on the same data set as the Theano DNN, using both cross-entropy and MPE sequence training to estimate the network weights. This system was evaluated on the broadcast news test set using Sphinx-4 with the trigram LM. Cross-entropy training yielded a 20.5% WER and sequence training yielded a 19.7% WER.

Next, the HTK-3.5 and Kaldi systems were evaluated on 56 hours of un-transcribed broadcast news audio. LM rescoring and N-best ROVER were applied as described above, and the automatically transcribed data were used to supplement the 76 hours of manually transcribed training data. One training partition was created using the full 132 hours audio, and three additional training partitions were created in an attempt to remove incorrectly recognized utterances from the automatically transcribed data. This was done using the word posterior probabilities obtained from N-best ROVER. First, the mean word posterior probability was calculated for each automatically transcribed utterance. Next, these utterances were sorted according to their mean word posterior probability. Lastly, partitions were created using the manually transcribed data and the top scoring 80%, 60%, and 40% of the automatically transcribed utterances. DNNs were then trained on each partition

using Theano;[18] all networks used the same architecture as the previously developed DNNs. Table 9 shows the WERs obtained on the broadcast news test set. Note that sequence training was not applied to these DNNs. The best WER was obtained using the top scoring 60% of the automatically transcribed utterances.

Table 9: Russian WER on the broadcast news test set using manually transcribed and automatically transcribed training data

All Systems were trained using Theano and a modified version of HTK-3.4.1; sequence training was not applied in this set of experiments.

Training Data	Hours	Decode	4-gram+RNN
manual	76	20.5	19.6
manual + all automatic	132	19.5	18.8
manual + top 80% automatic	127	19.6	18.7
manual + top 60% automatic	119	19.3	18.6
manual + top 40% automatic	108	19.7	18.8

2.1.7. Ukrainian

This section describes the Ukrainian ASR system that was developed for Haystack. Acoustic models were trained on 45 hours of broadcast news that were manually transcribed in the SCREAM Laboratory. A GMM-HMM speech recognition system, i-vector extractor, and hybrid DNN-HMM system were developed using the same procedure described in Section 2.1.3. The final HMM set included 3000 shared states, and the DNN included 5 hidden layers with 1024 ReLUs per layer.

Interpolated trigram and 4-gram LMs were estimated on the broadcast news transcripts, articles downloaded from Wikipedia,[19] and news articles downloaded from the internet. A maximum entropy RNN LM was trained on the same data set using the RNNLM toolkit. The network included 160 hidden units, 300 classes in the output layer, 4-gram features for the direct connections, and a hash size of 10^9. The vocabulary had 400000 words and was chosen using the select-vocab tool from the SRILM toolkit. A pronunciation dictionary was manually created for the most frequent 2000 words from the broadcast news corpus. This dictionary was then used to train a Sequitur grapheme-to-phoneme system for deriving the remaining pronunciations.

Next, a multilingual GMM-HMM system, i-vector extractor, and hybrid DNN-HMM system were

[18] These networks were trained using an update version of the Theano training script that applies a different training vector randomization scheme; this reduced the WER of the baseline system from 20.8% to 20.5%, which is the same WER as the HTK-3.5 system prior to sequence training

[19] Available at: http://dumps.wikipedia.org/ukwiki

trained on both the Ukrainian and Russian broadcast news data. The pronunciation dictionary used language-dependent phonemes, but allowed HMM states from different phonemes to be clustered together when building the decision tree.[20] The final HMM set included 5000 shared states, and the DNN included 5 hidden layers with 1024 ReLUs per layer. Table 10 shows the WERs obtained with the monolingual and multilingual acoustic models.

Table 10: Ukrainian WER on the broadcast news test

Training Data	Decode	4-gram	4-gram+RNN
Ukrainian	30.5	30.4	28.2
Ukrainian + Russian	29.9	29.9	27.5

A hybrid DNN-HMM system was also developed using HTK-3.5. The HNTrainSGD tool was used to train a DNN on the Ukrainian data using both cross-entropy and MPE sequence training. This system was evaluated on the broadcast news test set using HDecode with the trigram LM. Cross-entropy training yielded a 29.7% WER and sequence training yielded a 29.0% WER. Rescoring the sequence-trained system with the 4-gram and RNN reduced the WER to 26.9%. Kaldi ASR systems were developed using the same procedure described in Section 2.1.2. One DNN system was trained on MFCC features, and a second DNN system was trained on bottleneck features. Each system was evaluated on the broadcast news test set and the recognition lattices were rescored using the 4-gram and RNN LM. The DNN system trained on MFCC features yielded a 28.0% WER, and the DNN system trained on bottleneck features yielded a 26.5% WER. N-best ROVER was used to combine the output from the HTK-3.5 system and the two Kaldi system; this produced a 24.9% WER.

2.2 Haystack MMIER System

This section describes improvements made to the Haystack MMIER System. Section 2.2.1 discusses improvements made to the user interface. Section 2.2.2 discusses several improvements that were made to the processing pipeline.

2.2.1. User Interface Improvements

This section describes improvements made to the Haystack MMIER user interface with a new administrative section, updates to its file upload capabilities, inclusion of named entity detection, integration of OCR to the pipeline, and the addition of a HTML5-based media player.

[20] HMM state clustering was performed using the HHEd tool from HTS

Administration: An Administration page was developed to allow addition of new users and groups as well as assigning users, permissions, and roles to the groups. This capability enables multiple groups with an assigned administrator that controls the user additions and permissions per group, effectively making the Haystack MMIER System a portal. Several site-specific tools were also added to the Administration page. There is a section for minor editing of processed files and another for checking which ports have available Machine Translation (MT) servers. It is now possible to view error logs, access logs, logins, jobs, and searches. While testing newer versions of software such as SOLR, the administrator can swap the configured ports for testing.

File Upload: The Haystack System has had the capability to upload single files or a small number of files through the default file upload capability and multi-file upload capability called FileStack. However, when one wants to upload a large number of files, the FileStack system can be cumbersome, requiring the user to manually populate a number of fields for each file. To simplify multiple file uploads, a Bulk Directory Upload System was created to allow users to select a directory full of files to batch process.

For Bulk Directory Upload, required metadata fields are populated with reasonable default values. The source field for all of the files is set to be the name of the directory to which the application is pointed. If the directory, for example, is named Bulk_Upload_Aug10_2016, then that name will be used as a default value for the source field in the metadata for all of the files. The user can also input a new name to be applied for all of the files. The title to submit to the database is the filename with the file suffix removed. Often, a directory will contain files that are all of the same source language. If this is the case, the user can select the source language from a dropdown menu (as well as a topic/domain if available). If the files contain various languages, the user can select "Unknown," and Haystack will automatically identify the language for each specific file and translate it accordingly.

Named Entity Detection: Haystack incorporates two Named Entity Detectors (NED) in the pipeline, one is a Government-off-the-shelf (GOTS) product, and the other is the open-source MIT Information Extraction (MITIE) system.[21] A script was created to format the XML rendered by these detectors into a format accessible by the caption windows for display during media playback. A basic floating menu system was created to enable the user to choose between seeing the results of each detector during playback or to switch them off entirely.

OCR Integration: Google acquired the Tesseract OCR[22] system and released it as open source to the community. Tesseract has been integrated into Haystack for almost all of the supported languages. When Tesseract begins its OCR process, it segments one or more subimages of each image on which to perform OCR. The subimage file names, the recognized text, and original image coordinates of the subimages are saved into XML files. The captured text for each subimage is separately sent to MT in order to keep the text output parallel. Code was written to parse the XML files and rebuild the images into a browser page with the source text and translated text side-by-side. The interface allows the user to view the output as a complete image with translated text with a

[21] Available at: https://github.com/mit-nlp/MITIE

[22] Available at: https://github.com/tesseract-ocr

zoom tool for viewing a magnified version of the subimage and its corresponding OCR results. A potential future add-on to this viewer could allow a linguist to correct any anomalies he/she might see between the image and the text and to resubmit it for translation.

HTML5 Media Player: Previous versions of the Haystack Media Player used the Adobe Flash Player[23] for video and audio playback and for dynamic caption highlighting. Modern web browsers have started adopting HTML5 specifications with support for audio and video playback.

The Haystack pipeline was updated to generate MP4 and OGG audio and video formats. OGG and MP4 are approved HTML5 formats that provide cross-browser compatibility. Previously, all files were converted into the FLV format for Flash. In addition to removing an extra dependency on a third-party application, the move to HTML5 for audio and video playback has improved the overall load and response times for media playback.

A function was developed to update highlighting within text windows (utterance, translation, etc.) every second during media playback. If the highlighted section of text does not contain results from the current timestamp of the media being played, the boundaries of the current text are located, highlighted and scrolled to the top of the window. HTML5 allows for subtitles to print directly to the viewport of the playing media via an XML file. The user can decide which MT subtitles to display onscreen and dynamically change those options during playback. A potential future update will allow full screen media playback with onscreen subtitles.

2.2.2. Pipeline Improvements

Several improvements were made to the Haystack processing pipeline. Major additions include support for the following: hybrid DNN-HMM speech recognition systems with i-vector input, speech recognition using Kaldi, ASR system combination using N-best ROVER, OCR using Tesseract, named entity detection using MITIE, machine translation using BBN BOLT and SDL engines, and text recasing using Moses.

In order to support hybrid DNN-HMM systems with i-vectors, the Haystack ASR pipeline was updated to perform speaker change detection, speaker clustering, and speaker recognition prior to evaluating the first pass speech recognizer.[24] Speaker specific i-vectors are then extracted using HTK features and the MIT LLSpeech software described in Section 2.1.1.

Support was added to the pipeline for decoding multiple Kaldi speech recognition systems in parallel with Sphinx-4. The recognition lattices from these systems are rescored using the following procedure. First, the lattices are converted to HTK format and rescored with an N-gram LM. N-best lists are then extracted from each lattice and rescored with an RNN LM. The final LM score for each hypothesis is obtained by linearly interpolating the log probabilities from each model, and the best scoring hypothesis from each N-best list is selected for each utterance. The SRILM toolkit is used to extract the N-best lists and apply N-gram rescoring; the RNNLM toolkit is used for RNN rescoring.

[23] https://get.adobe.com/flashplayer

[24] The pipeline was originally designed to perform these processes after evaluating the first pass speech recognizer

ASR system combination is performed using N-best ROVER from the SRILM toolkit. Support is provided for combining the outputs from an HTK hybrid DNN-HMM system and one or more Kaldi ASR systems. Haystack currently performs system combination for English, Russian, and Ukrainian. As additional ASR systems become available, new languages can be added by creating a configuration file that specifies the N-best ROVER parameters.

OCR was integrated into the pipeline using Tesseract, which includes support for more than 90 languages. In addition to processing standard image files, the Portable Document Format (PDF) text extraction code was updated to handle files that contain both text and images. Layout information and raw text are extracted using PDFMiner;[25] images are extracted using the Linux utility pdfimages. Each image is processed using Tesseract and the recognized text is inserted at the appropriate position using the layout information from PDFMiner.

MITIE models were trained to detect named entities in Chinese, English, French, German, Portuguese, Spanish, and Russian. The Chinese models were trained on data from IWSLT 2015 and the Automatic Content Extraction (ACE) corpus [30]. Chinese text was tokenized using the Stanford Segmenter,[26] and the following named entities were tagged: Person, Location, Organization, and Geo-Political.

Models for the remaining languages were trained on the WikiNER corpus [31]; this corpus includes tags for Person, Location, Organization, and Miscellaneous. The Haystack pipeline was updated to evaluate MITIE on both the source and translated texts.

The BBN BOLT[27] and SDL[28] translation engines were integrated into the processing pipeline. BBN BOLT provides support for Arabic-to-English and Chinese-to-English; SDL includes support for Arabic-to-English and Russian-to-English.

Moses text recasers were developed for English, French, German, Portuguese, and Russian. These systems were trained on the following data sets: (1) English Gigaword, News Crawl 2007–2013, and TED; (2) French Gigaword [32], News Crawl 2007–2014, and Wikipedia; (3) German Euronews, News Crawl 2007–2014, and Wikipedia; (4) Portuguese CETEM Público [33], Portuguese News 1999 [34], and Wikipedia; and (5) Russian News Crawl 2008–2014 and Wikipedia.

Improvements to the Haystack pipeline include the following. First, dependencies on the International Computer Science Institute (ICSI) speech tools and Tcl programming language were replaced with C programs. Next, ZeroMQ translation servers were developed to queue documents for each translation engine. The queueing was done to improve stability by ensuring that each translation engine instance only processes a single document at a time. Finally, the audio segmentation process was updated to remove long periods of non-speech.

[25] Available at: http://pypi.python.org/pypi/pdfminer

[26] Available at: http://nlp.stanford.edu/software/segmenter.shtml

[27] http://www.opencatalog.darpa.mil/BOLT.html

[28] http://www.sdl.com

3.0 CONCLUSIONS

In conclusion, work has been accomplished in the areas of ASR and information extraction, especially in the context of the Haystack MMIER System.

For English ASR, the following DNN input features were investigated: PLP features, PLP features transformed using CMLLR, filterbank features, ALIZE i-vectors with z-score and EFR normalization, and LLSpeech i-vectors with EFR normalization. The best performance was obtained using filterbank features and LLSpeech i-vectors with EFR normalization. Improvements were also obtained by using ReLUs instead of logistic sigmoid units, and performing DNN speaker adaptation by updating the network with a single epoch of training and a small learning rate. An English ASR system was developed for the IWSLT 2015 evaluation that yielded a 6.6% WER on the tst2015 partition, and placed first out of the six ASR systems that were submitted for the evaluation. Arabic, Chinese, Farsi, Russian, and Ukrainian hybrid DNN-HMM speech recognition systems were developed for Haystack. The following methods were investigated for improving these systems: using Kaldi pitch features for Chinese, applying MPE sequence training using HTK-3.5, supplementing the Russian acoustic data with automatically transcribed broadcast news data, training multilingual DNNs on Russian and Ukrainian acoustic data, and combining the outputs from HTK and Kaldi systems using N-best ROVER.

Major additions to the Haystack user interface include the following: a new administrative section, updates to the file upload capability, inclusion of named entities, integration of OCR results, and the addition of an HTML5-based media player. The processing pipeline was updated to provide support for hybrid DNN-HMM speech recognition systems with i-vector input, speech recognition using Kaldi, ASR system combination, OCR using Tesseract, named entity detection using the MITIE tool, machine translation using BBN BOLT and SDL engines, and text recasing using Moses.

4.0 REFERENCES

[1] N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-End Factor Analysis for Speaker Verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, pp. 788–798, 2011.

[2] P. Kenny, G. Boulianne, and P. Dumouchel, "Eigenvoice Modeling with Sparse Training Data," *IEEE Transactions on Speech Audio Processing*, vol. 13, pp. 345–354, 2005.

[3] A. Larcher, J.-F. Bonastre, B. Fauve, K. A. Lee, C. Levy, H. Li, J. Mason, and J.-Y. Parfait, "ALIZE 3.0 – Open Source Toolkit for State-of-the-Art Speaker Recognition," in *Proceedings of Interspeech*, Lyon, France, 2013.

[4] M. Kazi, M. Coury, E. Salesky, J. Ray, W. Shen, T. Gleason, T. Anderson, G. Erdmann, L. Schwartz, B. Ore, R. Slyh, J. Gwinnup, K. Young, and M. Hutt, "The MIT-LL/AFRL IWSLT-2013 MT system," in *Proceedings of the International Workshop on Spoken Language Translation*, Heidelberg, Germany, 2013.

[5] S. Young *et al*. (2009) The HTK book. Cambridge University Engineering Department. [Online]. Available: http://htk.eng.cam.ac.uk

[6] G. Dahl, D. Yu, L. Deng, and A. Acero, "Context-Dependent Pre-trained Deep Neural Networks for Large Vocabulary Speech Recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, pp. 30–42, January 2012.

[7] G. Saon, J. Soltau, D. Nahamoo, and M. Picheny, "Speaker Adaptation of Neural Network Acoustic Models using I-Vectors," in *Proceedings of the Automatic Speech Recognition and Understanding Workshop (ASRU)*, Olomouc, Czech Republic, 2013.

[8] M. Kazi, E. Salesky, B. Thompson, J. Ray, M. Coury, W. Shen, T. Anderson, G. Erdmann, J. Gwinnup, K. Young, B. Ore, and M. Hutt, "The MITLL-AFRL IWSLT 2014 MT system," in *Proceedings of the International Workshop on Spoken Language Translation*, Lake Tahoe, CA, 2014.

[9] F. Seide, G. Li, and D. Yu, "Conversational Speech Transcription Using Context-Dependent Deep Neural Networks," in *Proceedings of Interspeech*, Florence, Italy, 2011.

[10] R. Gretter, "Euronews: A Multilingual Speech Corpus for ASR," in *Proceedings of the 9th edition of the Language Resources and Evaluation Conference (LREC)*, Reykjavik, Iceland, 2014.

[11] D. Graff, J. Garofolo, J. Fiscus, W. Fisher, and D. Pallett. (1997) 1996 English Broadcast News Speech (HUB4). Linguistic Data Consortium. Philadelphia. [Online]. Available: https://www.ldc.upenn.edu

[12] J. Fiscus, J. Garofolo, M. Przybocki, W. Fisher, and D. Pallett. (1998) 1997 English Broadcast News Speech (HUB4). Linguistic Data Consortium. Philadelphia. [Online]. Available: https://www.ldc.upenn.edu

[13] L. Lamel, J. Gauvain, and G. Adda, "Lightly Supervised and Unsupervised Acoustic Model Training," *Computer Speech and Language*, vol. 16, pp. 115–129, 2002.

[14] R. Parker, D. Graff, J. Kong, K. Chen, and K. Maeda. (2011) English Gigaword Fifth Edition. Linguistic Data Consortium. Philadelphia. [Online]. Available: https://www.ldc.upenn.edu

[15] R. Moore and W. Lewis, "Intelligent Selection of Language Model Training Data," in *Association of Computational Linguistics Conference Short Papers*, Uppsala, Sweden, 2010.

[16] T. Mikolov, A. Deoras, D. Povey, L. Burget, and J. Čermocky, "Strategies for Training Large Scale Neural Network Language Models," in *Proceedings of the Automatic Speech Recognition and Understanding Workshop*, Hawaii, USA, 2011.

[17] M. Zeiler, M. Ranzato, R. Monga, M. Mao, K. Yang, Q. Le, P. Nguyen, A. Senior, V. Vanhoucke, J. Dean, and G. Hinton, "On Rectified Linear Units for Speech Processing," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Vancouver, BC, 2013.

[18] S. Zhang, H. Jiang, S. Wei, and L.-R. Dai, "Rectified Linear Neural Networks with Tied-Scalar Regularization for LVCSR," in *Proceedings of Interspeech*, Dresden, Germany, 2015.

[19] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *Journal of Machine Learning Research*, vol. 15, pp. 1929–1958, 2014.

[20] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The Kaldi Speech Recognition Toolkit," in *Proceedings of the Automatic Speech Recognition and Understanding Workshop*, 2011.

[21] J. Kong and D. Graff. (2005) TDT4 Multilingual Broadcast News Speech Corpus. Linguistic Data Consortium. Philadelphia. [Online]. Available: https://www.ldc.upenn.edu

[22] R. Parker, D. Graff, K. Chen, J. Kong, and K. Maeda. (2011) Arabic Gigaword Fifth Edition. Linguistic Data Consortium. Philadelphia. [Online]. Available: https://www.ldc.upenn.edu

[23] B. Ore, S. Thorn, D. Hoeferlin, R. Slyh, and E. Hansen. (2012) Foreign Language Analysis and Recognition (FLARe) Initial Progress.

[24] M. Bisani and H. Ney, "Joint-Sequence Models for Grapheme-to-Phoneme Conversion," *Speech Communication*, vol. 50, pp. 434–451, 2008.

[25] C. Chen, R. Gopinath, M. Monkowski, M. Picheny, and K. Shen, "New Methods in Continuous Mandarin Speech Recogniti in *Proceedings of the 5th European Conference on Speech Communication and Technology (EUROSPEECH)*, Rhodes, Greece, 1997.

[26] P. Ghahremani, B. BabaAli, D. Povey, K. Riedhammer, J. Trmal, and S. Khudanpur, "A Pitch Extraction Algorithm Tuned for Automatic Speech Recognition," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Florence, Italy, 2014.

[27] R. Parker, D. Graff, K. Chen, J. Kong, and K. Maeda. (2011) Chinese Gigaword Fifth Edition. Linguistic Data Consortium. Philadelphia. [Online]. Available: https://www.ldc.upenn.edu

[28] S. Huang, J. Liu, X. Wu, L. Wu, Y. Yan, and Z. Qin. (1998) 1997 Mandarin Broadcast News Speech and Transcripts (HUB4-NE). Linguistic Data Consortium. Philadelphia. [Online]. Available: https://www.ldc.upenn.edu

[29] T. Schultz, "GlobalPhone: A Multilingual Speech and Text Database Developed at Karlsuhe University," in *Proceedings of the International Conference on Spoken Language Processing*, Denver, CO, 2002.

[30] C. Walker, S. Strassel, J. Medero, and K. Maeda. (2006) ACE 2005 Multilingual Training Corpus. Linguistic Data Consortium. Philadelphia. [Online]. Available: https://www.ldc.upenn.edu

[31] J. Nothman, N. Ringland, W. Radford, T. Murphy, and J. Curran, "Learning Multilingual Named Entity Recognition from Wikipedia," *Artificial Intelligence*, vol. 194, pp. 151–175, 2012.

[32] D. Graff, A. Mendonca, and D. DiPersio. (2011) French Gigaword Third Edition. Linguistic Data Consortium. Philadelphia. [Online]. Available: https://www.ldc.upenn.edu

[33] D. Santos and P. Rocha. (2001) CETEMpublico. Linguistic Data Consortium. Philadelphia. [Online]. Available: https://www.ldc.upenn.edu

[34] J. Wright and D. Graff. (1999) Portuguese Newswire Text. Linguistic Data Consortium. Philadelphia. [Online]. Available: https://www.ldc.upenn.edu

5.0 LIST OF ACRONYMS & GLOSSARY

ACE	Automatic Content Extraction
ALIZE	Open-source software package for speaker recognition
ASR	Automatic Speech Recognition
BOLT	Broad Operational Language Translation
CER	Character Error Rate
CMLLR	Constrained Maximum Likelihood Linear Regression
CMU	Carnegie Mellon University
DBN	Deep Belief Network
DNN	Deep Neural Network
DP	Dynamic Programming
EFR	Eigen Factor Radial
FFmpeg	Cross-platform software for recording, converting, and streaming audio and video
FLARe	Foreign Language Analysis and Recognition
fMLLR	feature-space Maximum Likelihood Linear Regression
GALE	Global Autonomous Language Exploitation
GMM	Gaussian Mixture Model
GOTS	Government off the Shelf
HLDA	Heteroscedastic Linear Discriminate Analysis
HMM	Hidden Markov Model
HTK	Cambridge University Hidden Markov Model Toolkit
HUB4	Broadcast news corpus (text and audio) released by the Linguistic Data Consortium
ICSI	International Computer Science Institute
IWSLT	International Workshop on Spoken Language Translation
JFA	Joint Factor Analysis
kHz	Kilohertz
LDC	Linguistic Data Consortium
LM	Language Model
MFCC	Mel Frequency Cepstral Coefficient
MIT	Massachusetts Institute of Technology

MITIE	Massachusetts Institute of Technology Information Extraction
MMIER	Multilingual Multimedia Information Extraction and Retrieval
MPE	Minimum Phone Error
MT	Machine Translation
OCR	Optical Character Recognition
PDF	Portable Document Format
PDFMiner	Tool for extracting information from portable document format files
PLP	Perceptual Linear Prediction
ReLU	Rectified Linear Unit
RNN	Recurrent Neural Network
ROVER	Recognizer Output Voting Error Reduction
SAD	Speech Activity Detector
SAT	Speaker Adaptive Training
SCREAM	Speech and Communication Research, Engineering, Analysis, and Modeling
sMBR	state-level Minimum Bayes Risk
SoX	Sound Exchange Toolkit
SRILM	Language modeling toolkit developed at Stanford Research Institute
TDT	Topic Detection and Tracking
TED	Technology, Entertainment, and Design
TRANSTAC	Translation System for Tactical Use
UBM	Universal Background Model
WER	Word Error Rate
WMT	Association for Computational Linguistics Workshop on Machine Translation